基于 Transformer 的图像分类网络 MultiFormer *

胡 杰 a,b,c†，昌敏杰 a,b,c，熊宗权 a,b,c，徐博远 a,b,c，谢礼浩 a,b,c，郭 迪 a,b,c

(a.现代汽车零部件技术湖北省重点实验室; b.汽车零部件技术湖北省协同创新中心; c.湖北省新能源与智能网联车工程技术研究中心 武汉理工大学，武汉 430070)

摘 要：为解决目前 ViT 模型无法改变输入补丁大小且输入补丁都是单一尺度信息的缺点，提出了一种基于 Transformer 的图像分类网络称为 MultiFormer。MultiFormer 通过 AWS(Attention With Scale)模块，将每阶段不同尺度输入小补丁嵌入为具有丰富语义信息的大补丁；通过 GLA-P(Global-Local Attention With Patch)模块交替捕获局部和全局注意力，在嵌入时同时保留了细粒度和粗粒度特征。设计了 MultiFormer-Tiny、-Small 和-Base 三种不同变体的 MultiFormer 模型网络，在 ImageNet 图像分类实验中 Top-1 精度分别达到 81.1%、82.2%和 83.2%，后两个模型对比同体量的卷积神经网络 ResNet-50 和 ResNet-101 提升 3.1%和 3.4%；对比同样基于 Transformer 分类模型 ViT，MultiFormer-Base 在参数和计算量远小于 ViT-Base/16 模型且不需要大量数据预训练前提下提升 2.1%。

关键词：机器视觉；深度学习；图像分类；自注意力；Transformer

中图分类号：TP391.41 doi: 10.19734/j.issn.1001-3695.2022.03.0133

MultiFormer: image classification network based on Transformer

Hu Jie[a,b,c†], Chang Minjie[a,b,c], Xiong Zongquan[a,b,c], Xu Boyuan[a,b,c], Xie Lihao[a,b,c], Guo Di[a,b,c]

(*a. Hubei Key Laboratory of Advanced Technology for Automotive Components*, *b. Hubei Collaborative Innovation Center for Automotive Components Technology*, *c. Hubei Research Center for New Energy & Intelligent Connected Vehicle Wuhan University of Technology*, *Wuhan* 430070, *China*)

Abstract: In order to solve the disadvantage that the ViT cannot change the input patch size and the input patches are all single-scale information, this paper proposed an image classification network based on Transformer called MultiFormer. MultiFormer embeds small patches with different scales of input at each stage into large patches with rich semantic information through the AWS (Attention With Scale) module; and captures local and global attention alternately through the GLA-P (Global-Local Attention With Patch) module, preserving both fine-grained and coarse-grained features during embedding. This paper designed MultiFormer-Tiny, -Small and -Base networks of three different sizes to achieve 81.1%, 82.2% and 83.2% Top-1 accuracy respectively in ImageNet image classification experiments, the latter two models improve by 3.1% and 3.4% compared to the same volume of convolutional neural networks ResNet-50 and ResNet-101; MultiFormer-Base offers a 2.1% improvement with far fewer parameters and computational effort than the ViT-Base/16 model, and without the need for extensive data pre-training.

Key words: machine vision; deep learning; image classification; self-attention; Transformer

0 引言

一方面，图像分类[1]、目标检测[2]和语义分割[3]等计算机视觉任务由卷积神经网络主导，自 AlexNet[4]在 ImageNet 图像分类挑战中获得冠军之后，卷积神经网络架构通过一系列设计变得更深、更密集且卷积形式更复杂[5~7]，ResNet[5]提出了残差网络在加深网络层数时解决了梯度消失问题；DenceNet[6]引入了密集连接的拓扑结构将每个卷积块与前一个卷积块连接起来；VGG[8]通过叠加卷积核扩大感受野的方法加深网络；GoogLeNet[9]通过构建密集的块结构来近似最优的稀疏结构在提高性能时不增加计算量； EfficientNet[10]证明了可以利用复合系数统一缩放模型所有维度从而提高模型性能。另一方面，Transformer 由于自注意力模块具有捕捉长距离依赖[11]的能力而被用于自然语言处理任务，许多研究人员受此启发，尝试探索 Transformer 结构在计算机视觉任务中的应用。文献[12~15]已将自注意力模块纳入卷积神经网络并用于图像分类、目标检测和语义分割等计算机视觉任务。

Vision Transformer(ViT)[16]由于不使用卷积神经网络而通过图像序列化将 Transformer 应用于图像分类，因此迅速引入改进[17~20]并用于各种下游任务[21~24]。由于 Transformer 的自注意力模块对整个输入序列进行操作，处理自然图像时把每一个像素点都看做一个标记，其长度会远远长于单词序列，因此会比卷积操作产生更多的内存和计算成本。ViT 采用折中策略将多个像素点嵌入图像补丁(Patch)作为一个标记(Token)输入自注意力模块进行计算，但是计算复杂度仍然过高且要求输入图片只能是固定大小。对 ViT 的改进可以分为三类：

a)改进 ViT 设计本身，DeiT[19]引入了合适的训练策略来摆脱大规模的预训练并采用蒸馏的方式引导模型进行更好的学习；T2T-ViT[20]采用渐进式方式将图像结构化为图片补丁并保留了局部结构信息，克服了 ViT 中简单标记化的局限性；Dynamicvit[23]利用 Transformer 标记是非结构化序列的特点，

收稿日期：2022-03-28；**修回日期：**2022-05-17 **基金项目：**湖北省技术创新专项(2019AEA169)；湖北省科技重大专项(2020AAA001)

作者简介：胡杰(1984-)，男(通信作者)，湖南永州人，副教授，博导，博士，主要研究方向为智能网联汽车、车联网与大数据(auto_hj@163.com)；昌敏杰(1999-)，男，湖北荆州人，硕士研究生，主要研究方向为机器视觉；熊宗权(1995-)，男，江苏南京人，硕士研究生，主要研究方向为车道线检测；徐博远(1998-)，男，湖北仙桃人，硕士研究生，主要研究方向为目标检测；谢礼浩(1996-)，男，江苏徐州人，硕士研究生，主要研究方向为目标检测；郭迪(1996-)，男，湖南常德人，硕士研究生，主要研究方向为目标检测．

设计了一种标记稀疏化剪枝的方法，通过删除信息量不大的标记降低计算量。

b)将卷积操作引入到 ViT 设计中，利用卷积进行位置编码[17]或者使用卷积来替换 Transformer 中的线性投影层[25]；CoAtNet[26]通过引入卷积神经网络捕获局部注意力来弥补局部特征。

c)设计新的主干网络和自注意力模块，PVT[21]设计了一个金字塔结构的主干网络逐层对特征图进行下采样并使用了空间缩减注意力模块来权衡模型效率和准确率；CAT[24]设计了一个跨块自注意块将序列补丁内的注意力和序列补丁间的注意力结合起来从而使得局部信息和全局信息交互；Swin[22]提出将输入特征图划分到不同固定大小的局部窗口中，通过在每个窗口内计算自注意力来降低计算成本；DPT[27]自适应地将图像分割成不同位置和大小的像素块，可以避免对语义信息的破坏从而捕捉到完整且与对象相关的局部结构；FPT[28]能够对特征图跨空间和跨尺度的非局部特征进行编码且能整合到其他主干网络用于其他下游任务。

这些工作仍然具有一些局限性，它们将单一尺度的图片补丁输入自注意力模块时会丢失许多语义信息，此时需要跨尺度注意力机制来建立它们之间的联系，同时图像分类任务需要粗粒度特征和细粒度特征之间的交互来捕获目标信息。基于上述，本文主要工作如下：a)针对 ViT 模型输入特征图存在尺度单一的问题，提出多尺度嵌入模块 AWS(Attention With Scale)，AWS 模块使用不同尺度的卷积核对图片进行采样，融合成具有人类视觉感受野的跨尺度注意力补丁输入到下一个阶段，使每一个阶段的输入都是多尺度图片补丁；b)针对其他 Transformer 模型无法对像素块长距离建模导致粗粒度特征丢失的问题，设计新的自注意力模块 GLA-P(Global-Local Attention With Patch)，通过交替捕获全局补丁和局部补丁聚合粗粒度和细粒度特征来弥补图片补丁语义信息的不足，利用注意力打包操作，在不影响网络性能的前提下减少计算量；c)设计了三个不同大小的模型称为 MultiFormer-Tiny、-Small 和 -Base，在公开数据集 ImageNet、CIFAR10 和 CIFAR100 上进行图像分类实验，结果表明 MultiFormer 在图像分类实验中优于其他同量级的对比网络，并通过消融实验验证了各个模块的有效性。

1　模型框架

MultiFormer 图像分类网络整体框架如图 1(a)所示，主干网络参考 PVT[21]设计为 4 阶段金字塔结构，每个阶段由 AWS 多尺度嵌入模块和多个 MultiFormer Block 顺序组合而成；如图 1(b)所示，每个 MultiFormer 模块由一个 GLA-P 自注意力模块和一个多层感知机 MLP 组成，MLP 使输入数据非线性化并改变数据维度。

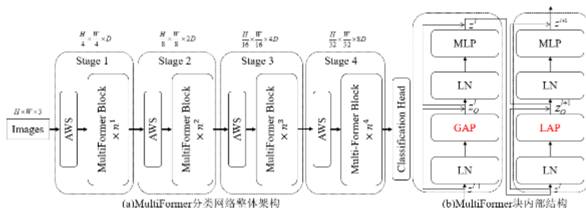


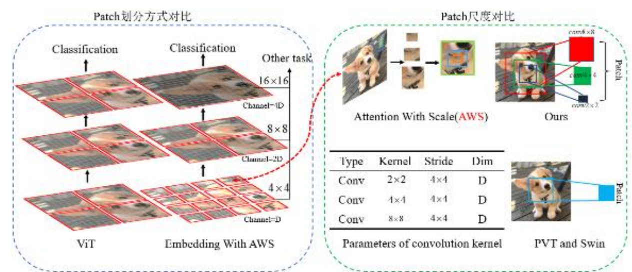
图 1　算法框架

Fig.1　Algorithm framework

首先，输入图片通过 AWS 模块生成多尺度特征图并划分为具有多尺度信息的图片补丁，对除了 Stage1 之外的输出进行下采样将补丁数量减少为四分之一并把输出维度扩大为两倍形成金字塔结构。然后，把生成的多尺度补丁接入 MultiFormer Block 中的 GLA-P 自注意力模块，GAP(Global Attention with Patch)和 LAP(Local Attention with Patch)即打包过的全局注意力和局部注意力交替出现形成 GLA-P 自注意力模块能够同时聚合输入图像的全局特征和局部特征。最后，接一个单独的视觉任务头如图像分类头(Classification Head)用于图像分类任务，下面详细介绍各个模块原理及作用。

1.1　AWS 多尺度嵌入模块

将图像输入自注意力模块计算之前，需要将图片的像素块划分为等大小的图片补丁并序列化为二维矩阵形式来满足输入要求。如图 2 中补丁 Patch 划分方式对比所示，ViT 简单地将图片中相邻的像素块划分为固定大小的补丁使每个阶段的补丁数量固定，从而方便嵌入绝对位置编码输入自注意力模块计算。这种划分方式会导致每个阶段的自注意力计算量呈平方倍增长，并需要大量数据集进行预训练而且难以训练到收敛。与 ViT 不同，AWS 模块先将图片划分成大小为 4×4 的小补丁，然后利用下采样将小补丁合并为 8×8 和 16×16 的大补丁并将维度升为 2 倍，通过减少补丁数量，扩大补丁维度和大小形成金字塔结构，不仅降低了计算复杂度，而且不必限制每个阶段的补丁数量，解决了 ViT 必须输入固定大小图片的劣势。



Type	Kernel	Stride	Dim
Conv	2×2	4×4	D
Conv	4×4	4×4	D
Conv	8×8	4×4	D

图 2　AWS 模块

Fig.2　Module diagram of AWS

PVT[21]和 Swin[22]在将输入图片划分为等大小的补丁并序列化为二维矩阵时，由于忽视了输入特征图尺度对图片补丁尺度的影响，使得划分的补丁尺度单一并会丢失目标的多尺度语义信息，导致模型性能降低。本文设计的 AWS 模块在生成图片补丁之前会利用不同大小的卷积核对输入图片进行卷积操作，生成语义信息丰富的多尺度特征图用于增强嵌入补丁的语义信息。如图 2 所示，首先，AWS 模块接收一个 $H\times W\times 3$ 的 RGB 图像作为输入，使用三个不同大小的卷积核进行采样，将采样框的步幅保持一致，让每个采样框都有相同的中心和不同的尺度，其中 Stage1 的 AWS 卷积核大小设置为 2×2、4×4 和 8×8，后三个 Stage 设置为 2×2 和 4×4，步幅都设置为 4×4，为了便于特征图的融合将通道数都设置为 D。然后，将通过不同尺度卷积核采样得到的多尺度特征图参考人眼视觉特征[29]融合成语义信息丰富的特征嵌入图。最后，将 Stage1 中划分的 4×4 大小的补丁下采样为 8×8 和 16×16 大小的补丁，并将维度扩大为两倍形成金字塔结构。与其他 Transformer 网络划分的 Patch 尺度对比如图 2 右侧所示，PVT 和 Swin 将输入图片进行划分时，粗糙地将原始特征图划分为图片补丁，此时的补丁受特征图尺度的限制只能关注到 4×4 像素块里的特征信息，如果目标尺度不局限于 4×4 大小的像素块之内，则模型会因为无法关注到图片目标其他尺度内的语义信息而造成目标特征信息缺失；本文提出的 AWS 模块通过多尺度卷积操作使得划分的补丁能够聚合特征图中 2×2、4×4 和 8×8 多个尺度中像素块的语义信息从而生成特征信息丰富的补丁，在输入后续模块时能够弥补 Swin 和 PVT 由于补丁多尺度特征信息不足而提升模型性能。

1.2　GLA-P 自注意力模块

通过 AWS 模块生成多尺度补丁之后，需要将图片补丁输入图 1(a)MultiFormer 模块中的 GLA-P 自注意力模块计算。如图 1(b)所示，由于在图像分类任务中，网络需要同时捕获目标的细粒度和粗粒度特征，因此在 MultiFormer 模块中设

计了 GAP 和 LAP 交替形成新的自注意力模块 GLA-P，从而能够捕获全局注意力和局部注意力来保留目标的粗粒度和细粒度特征。如图 3 所示，输入 LAP 和 GAP 的是经过 AWS 模块嵌入的多尺度特征图 $H_O \times W_O \times D$，对于 LAP，每 4×4 的相邻像素块被分组在一起形成 Local Attention；对于 GAP，同样 4×4 数量但间隔为 4 的像素块被分为一组形成 Global Attention，不相邻的像素块由于广泛分布为生成的补丁提供了足够的上下文信息，使得全局注意力变得更加有效。

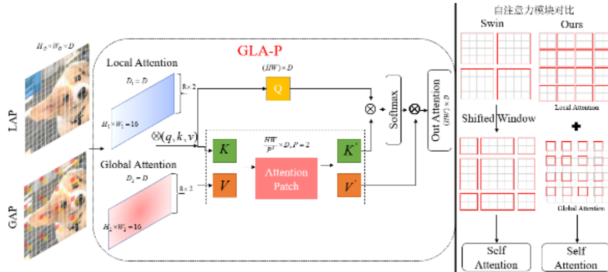


图 3　Global-Local Attention With Patch 模块

Fig.3　Module diagram of GLA-P

与 CoAtNet[26]中的 GLA(Global-Local Attention)通过卷积神经网络捕获局部特征，自注意力模块捕获全局特征不同，本文提出的 GLA-P 模块通过对相邻和相间隔的像素块分别进行短距离和长距离建模而不依赖于卷积神经网络来交替捕获全局注意力和局部注意力。为了更直观地观察 GLA-P 自注意力模块的作用方式，以 MultiFormer-Base 为例，将训练好的模型最后一层特征图输出的各个像素得分经过激活函数后，映射回原图得到 Global-Local Attention 自注意力可视化表述，如图 4 所示，明亮部分为自注意力所关注的部位，说明本文自注意力模块能有效捕获图像全局信息。

与其他 Transformer 模型的自注意力模块对比如图 3 所示，Swin 将特征图划分为几个不重叠的窗口并限制在每一个窗口内独立执行自注意力操作，此外为了补偿缺失的全局信息，提出了一种滑动窗口策略在不同窗口之间交换信息，不过 Swin 依然将自注意力计算局限在相邻的像素块之间，无法对生成的补丁进行长距离建模。本文提出的自注意力模块通过 Global Attention 对广泛分布的像素块进行建模而生成具有上下文信息的补丁，与通过 Local Attention 对相邻像素块建模生成的补丁相结合，经过 GLA-P 形成的补丁同时保留输入图片的全局信息和局部信息，在自注意力计算后能够同时关注目标的粗粒度和细粒度特征从而能够在图像分类任务中表现出色。



图 4　自注意力可视化

Fig. 4　Self-attention visualization

实际上，为了尽可能保留原特征图的语义信息，本文大补丁的分辨率会比较大(如 Stage1 中经过 GLA-P 处理后的补丁大小为 28×28)，在序列化为二维数组时计算量依然会很大，因此本文设置了一个卷积打包方式来替代传统编码器中的多头注意力(Multi-head attention)[11]，与 MHA 类似，接收查询(Query)、键(Key)和值(Value)并输出一个改进的加强特征。细节表述如下：

$$Patch(q,k,v,P=2,B)=Attention(Q,K^{'},V^{'},B) \tag{1}$$

$$\begin{cases} Q = xq \\ K^{'} = Norm(reshape(Conv(xk))) \\ V^{'} = Norm(reshape(Conv(xv))) \end{cases} \tag{2}$$

其中，x 为输入特征图，$q,k,v \in \mathbb{R}^{(HW) \times D}$ 为生成的对应维度的矩阵，本文在注意力每个头部都添加一个可学习的相对位置偏差[30~32] $B \in \mathbb{R}^{d_{head} \times (WH) \times (WH/d_{head}/P)}$，$Conv$ 为对应的卷积操作，与 $Patch(q,k,v,P=2)$ 中 P 的大小有关，例如 $P=2$，则卷积核大小为 4，$K^{'}$ 和 $V^{'}$ 的维度为 Q 的四分之一，$Norm(\cdot)$ 为层归一化[33]，$Attention(\cdot)$ 为自注意力操作，计算如下：

$$Attention(q,k,v)=SoftMax(\frac{qk^{T}}{\sqrt{d_{head}}}+B)v \tag{3}$$

与 Swin 同时处理 Query、Key 和 Value 不同，本文对键值对进行下采样后能在不影响精度的情况下减少 P 倍计算量，经过 Attention Patch 打包操作后参数减少情况如表 1 所示。

表 1　模型参数及计算量示意

Table 1　Model parameters and calculation quantity

Model		MultiFormer-Tiny	MultiFormer-Small	MultiFormer-Base
Before	Params(M)	32.7	39.8	54.7
Patch	Flops(G)	3.2	5.4	7.4
After	Params(M)	22.8	30.5	45.7
Patch	Flops(G)	2.4	4.8	6.9

同时由图 2 可将 MultiFormer 计算细节描述如下：

$$\begin{cases} z_O^{l} = GAP(LN(z^{l-1})) + z^{l-1} \\ z^{l} = MLP(LN(z_O^{l})) + z_O^{l} \\ z_O^{l+1} = LAP(LN(z^{l})) + z^{l} \\ z^{l+1} = MLP(LN(z_O^{l+1})) + z_O^{l+1} \end{cases} \tag{4}$$

其中，z_O^{l} 和 z^{l} 表示 MultiFormer 块中 GAP 模块和 MLP 模块的输出特征，z_O^{l+1} 和 z^{l} 表示 LAP 模块和 MLP 模块的输出特征。

1.3　模型变体

遵循残差网络结构 ResNet[5]的设计规则，本文构建了三个不同尺度大小的模型，分别称为 MultiFormer-Tiny，-Small，和-Base，它们的模型大小和计算复杂度为 1:1.5:3 的关系，其中 MultiFormer-Tiny、-Small 和-Base 的计算量和计算参数分别与 ResNet-18、ResNet-50 及 ResNet-101 相似，主要超参数设置如下：

MultiFormer-Tiny: $D=64, Depth=\{1,1,8,6\}, Heads=\{2,4,8,16\}$

MultiFormer-Small: $D=96, Depth=\{2,2,6,2\}, Heads=\{3,6,12,24\}$

MultiFormer-Base: $D=96, Depth=\{2,2,12,2\}, Heads=\{3,6,12,24\}$

其中，D 为第一阶段隐藏层的通道数，$Depth$ 为每个 stage 包含的 MultiFormer 块数，$Heads$ 为多头注意力的维度。且在 GLA-P 模块中将小补丁嵌入为大补丁时的大小为 $Group_size=\{28,14,14,7\}$，利用卷积将注意力打包时将 Patch 设置为 $P=\{4,2,2,1\}$，模型设计及详细超参数设置如表 2 所示。

2　实验

用本文设计的 MultiFormer 图像分类网络在 ImageNet-1K、CIFAR10 和 CIFAR100 数据集上进行图像分类实验并与同量级且具代表性的卷积神经主干网络 ResNet[5]以及其他基于 Transformer 的主流模型进行对比，随后进行充分的消融实验验证各个模块的有效性。

2.1　图像分类实验

ImageNet-1K 数据集[34]包含来自 1000 个类别的 128 万张训练图片和 5 万张验证图片，本文在训练集上训练模型，并用验证集测试输出 Top-1 精确度(排名第一的类别与实际结果相符的准确率)。本文将图像大小随机裁剪为 224×224，优化器选择动量为 0.9 且衰减权重为 0.05 余弦衰减的 AdamW 优化器，批次(batch_size)设为 128，初始学习率为 0.001，所有模型都在 4 张 2080Ti 显卡上从头开始训练 300 个 epoch，实验结果如表 3 所示。

表 2　MultiFormer 主干网络模型变体

Tab. 2　Model variants of backbone networks of multiformer

Output Size	Layer Name	MultiFormer-T	MultiFormer-S	MultiFormer-B
Stage-1	AWS		Kernel size：$4\times4,8\times8,16\times16, Stride=4$	
56×56	GLA-P	$\begin{bmatrix} D_1=64, H_1=2 \\ G_1=28, P_1=4 \end{bmatrix}\times1$	$\begin{bmatrix} D_1=96, H_1=3 \\ G_1=28, P_1=4 \end{bmatrix}\times2$	$\begin{bmatrix} D_1=96, H_1=3 \\ G_1=28, P_1=4 \end{bmatrix}\times2$
	MLP			
Stage-2	AWS		Kernel size：$2\times2,4\times4, Stride=2$	
28×28	GLA-P	$\begin{bmatrix} D_1=128, H_1=4 \\ G_1=14, P_1=2 \end{bmatrix}\times1$	$\begin{bmatrix} D_1=192, H_1=6 \\ G_1=14, P_1=2 \end{bmatrix}\times2$	$\begin{bmatrix} D_1=192, H_1=6 \\ G_1=14, P_1=2 \end{bmatrix}\times2$
	MLP			
Stage-3	AWS		Kernel size：$2\times2,4\times4, Stride=2$	
14×14	GLA-P	$\begin{bmatrix} D_1=256, H_1=8 \\ G_1=14, P_1=2 \end{bmatrix}\times8$	$\begin{bmatrix} D_1=384, H_1=12 \\ G_1=14, P_1=2 \end{bmatrix}\times6$	$\begin{bmatrix} D_1=384, H_1=12 \\ G_1=14, P_1=2 \end{bmatrix}\times12$
	MLP			
Stage-4	AWS		Kernel size：$2\times2,4\times4, Stride=4$	
7×7	GLA-P	$\begin{bmatrix} D_1=512, H_1=14 \\ G_1=7, P_1=1 \end{bmatrix}\times6$	$\begin{bmatrix} D_1=768, H_1=24 \\ G_1=7, P_1=1 \end{bmatrix}\times2$	$\begin{bmatrix} D_1=768, H_1=24 \\ G_1=7, P_1=1 \end{bmatrix}\times2$
	MLP			
Head	Avg Pooling		Kernel size：7×7	
1×1	Linear		Classes：1000	

表 3　分类实验 Top-1 精度对比

Tab. 3　Comparison of top-1 accuracy of classification experiment

Method	Param/M	FLOPs/G	Top-1/%
R18*[5]	11.7	1.8	69.8
R18[5]	11.7	1.8	68.5
DeiT-T[19]	5.7	1.3	72.2
PVT-S[21]	24.5	3.8	79.8
MultiFormer-Tiny	22.8	2.4	81.1
R50*[5]	25.6	4.1	76.1
R50[5]	25.6	4.1	78.5
DeiT-S[19]	22.1	4.6	79.9
T2T-ViT[20]	21.5	5.2	80.7
Swin-T[22]	29.0	4.5	81.3
CAT-S[24]	37.0	5.9	81.8
PVT-M[21]	44.2	6.7	81.2
MultiFormer-Small	30.5	4.8	82.2
R101*[5]	44.7	7.9	77.4
R101[5]	44.7	7.9	79.8
Swin-S[22]	50.0	8.7	83.0
CAT-B[24]	52.0	8.9	82.8
PVT-L[21]	61.4	9.8	81.7
DeiT-B[19]	86.0	17.5	81.1
ViT-Base/16[16]	86.6	17.6	81.1
MultiFormer-Base	45.7	6.9	83.2

从表 3 结果可以看出本文所设计的 MultiFormer 网络模型在参数量和计算量相当的情况下明显优于基于卷积神经网络的模型 ResNet 系列，MultiFormer-Tiny，-Small 和-Base 模型较 ResNet-18，ResNet-50 和 ResNet-101 模型分别提升 12.6%，3.1%和 3.4%；对比同样基于 Transformer 的主流模型 ViT 和 Swin，在参数和计算量远小于 ViT-Base/16 模型且不需要大量数据预训练前提下，MultiFormer-Base 提升 2.1%，同时在参数量较 Swin-S 降低了 10%的前提下提升 0.2%，验证了所设计模型的有效性。

图 5 为本文所设计的 MultiFormer 网络模型与卷积神经网络 ResNet 以及其他基于 Transformer 工作的网络模型对比，图 5(a)和(b)分别为模型参数和模型计算量与分类数据集 Top-1 准确率的关系，可以看出本文所设计的 MultiFormer 网络模型在参数量和计算量相当的情况下全面优于其他模型。

随机从 ImageNet 数据集中抽取图片，输入已加载训练权重的 MultiFormer、PVT 和 Swin 网络中进行推理，将 4 个阶段所得到的特征图相加并映射回原图得到图像分类实验热力图如图 6 所示。相较于 PVT 和 Swin 网络，MultiFormer 在处理单一尺度图片时，由于自注意力模块 GAL-P 能对上下文关系建模，因此能更加聚焦于目标的有效特征；在处理多尺度图片时，由于 AWS 多尺度嵌入模块能生成语义信息丰富的多尺度补丁，因此能有效关注目标的不同尺度信息及其轮廓信息。

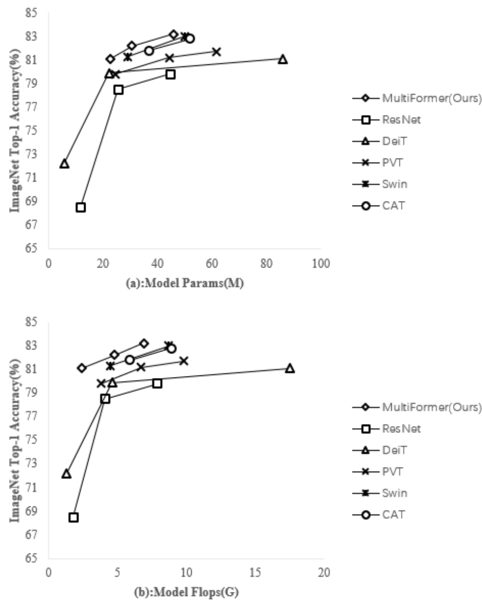


图 5　实验结果折线对比

Fig. 5　Broken line comparison diagram of experimental results

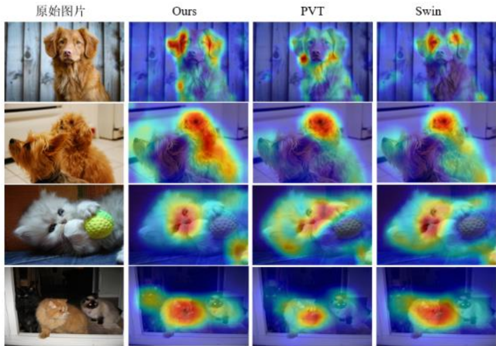


图 6　图像分类热力图对比图

Fig. 6　Thermodynamic diagram comparison of Image classification

使用 CIFAR10 和 CIFAR100 图像分类数据集对所设计的 MultiFormer 网络作进一步验证，CIFAR10 和 CIFAR100 分别包含 10 个和 100 个类别，两个数据集都分别有 5 万张训练集和 1 万张测试集，在训练集上训练模型，并用验证集测试输出 Top-1 精确度。为了避免由于数据集较小而出现过拟合的情况，与 ViT 微调策略保持一致，将 ImageNet 分类实验获得的训练权重分别加载到 MultiFormer-Tiny、-Small 和-Base 中并替换掉分类检测头，使用动量为 0.9 的 SGD 优化器进行模型微调，训练批次和轮数设置为 64 和 300 轮，实验结果如表 4 所示。

表 4 CIFAR 实验结果表

Tab. 4 Experimental results of CIFAR

Model	Param/M	CIFAR10	CIFAR100
EfficientNetV2-S[10]	24	98.7	91.5
EfficientNetV2-M[10]	55	99.0	92.2
EfficientNetV2-L[10]	121	99.1	92.3
LeViT-256[35]	18.9	98.0	—
LeViT-384[35]	39.1	98.1	—
ViT-B/16[16]	86	98.9	91.6
ViT-L/16[16]	307	99.1	93.4
ViT-H/16[16]	632	99.2	93.8
MultiFormer-Tiny	22.8	99.0	92.2
MultiFormer-Small	30.5	99.4	93.8
MultiFormer-Base	45.7	99.5	94.1

由表 4 可知，MultiFomer-Base 在参数量较 EfficientNetV2-L 降低了 50%的前提下，CIFAR10 和 CIFAR100 的 Top-1 精度分别提高 0.4%和 1.7%，在参数量为 ViT-H/16 的十分之一时，CIFAR10 和 CIFAR100 的 Top-1 精度仍能分别提高 0.3%和 0.3%；MultiFormer-Tiny 和-Small 对比同体量模型 LeViT-256 和-384 在 CIFAR10 上 Top-1 精度分别提高 1.0%和 1.3%，进一步验证了本文所提模型 MultiFormer 的有效性。

2.2 消融实验

为了验证所设计的 AWS 和 GLA-P 模块的有效性，本文基于 MultiFormer-Tiny 模型在 ImageNet 图像分类数据集上设计了如下对比实验。

a)取消 AWS 多尺度嵌入模块，改为单尺度嵌入，将 Stage-1 中卷积核设为单个大小为 4×4 的卷积核，其他阶段的下采样设为单个大小为 2×2 的卷积核，结果如表 5 所示，AWS 模块帮助模型取得了很大的性能提升，Top-1 准确率相较于单尺度嵌入提升了 0.6%。

b)用 GLA-P 模块替换为 Swin[22]、PVT[21]和 CoAtNet[26]模型中的自注意力模块，结果显示精度分别提升 0.3%、0.5%和 0.8%，具体分析是因为 Swin 采用了滑动窗口的方式将自注意力限制在了局部范围而忽略掉了全局注意力之间的联系；PVT 在处理自注意力特征时，对生成的键值对简单下采样而舍弃掉了细粒度语义信息；CoAtNet 在主干网络前两个阶段过度依赖卷积神经网络提取特征会丢失部分全局信息，导致输入图片粗粒度特征的缺失而精度降低。以上实验结果表明交替捕获局部注意力和全局注意力能有效提升模型性能。

表 5 消融实验

Tab. 5 Ablation experiment

AWS	GLA-P	Swin-Attention	PVT-Attention	CoAtNet	Top-1/%
	✓				80.5
✓		✓			80.8
✓			✓		80.6
✓				✓	80.3
✓	✓				**81.1**

实验条件和超参数均与之前保持一致，训练设备均为 4 张 2080Ti 显卡，训练轮数为 300 轮。

3 结束语

本文提出了一种基于 Transformer 的图像分类网络 MultiFormer，核心组成为(AWS)Attention With Scale 多尺度嵌入模块和(GLA-P)Global-Local Attention With Patch 自注意力模块，实验结果表明在参数和计算量相当的情况下相对于卷积神经网络和其他基于 Transformer 的工作有较大提升，证明了多尺度嵌入和交替捕获局部注意力及全局注意力能明显增强 Transformer 网络中自注意力学习特征图语义信息的能力，同时本文所设计的主干网络能较好提取特征图的语义信息，有望成为计算机视觉任务通用主干网络并用于其他下游任务。目前 Transformer 正在计算机视觉领域飞速发展并成为了一种趋势，希望本文能对后续基于 Transformer 模型所进行的工作能够起到启迪作用。

参考文献：

[1] 黄凯奇，任伟强，谭铁牛. 图像物体分类与检测算法综述 [J]. 计算机学报, 2014, 378 (06): 1225-1240. (Huang Kaiqi, Ren Weiqiang, Tan Tieniu. A review on image object classification and detection [J]. Chinese Journal of Computers, 2014, 378 (06): 1225-1240.)

[2] 李旭冬，叶茂，李涛. 基于卷积神经网络的目标检测研究综述 [J]. 计算机应用研究, 2017, 312 (10): 2881-2886, 2891. (Li Xudong, Ye Mao, Li Tao. Review of object detection based on convolutional neural networks [J]. Application Research of Computers, 2017, 312 (10): 2881-2886, 2891.)

[3] 田萱，王亮，丁琪. 基于深度学习的图像语义分割方法综述 [J]. 软件学报, 2019, 30 (02): 440-468. (Tian Xuan, Wang Liang, Ding Lei. Review of image semantic segmentation based on deep learning [J]. Journal of Software, 2019, 30 (02): 440-468.)

[4] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks [C]// Advances in Neural Information Processing Systems. 2012: 1097–1105.

[5] He Kaiming, Zhang Xiangyu, Ren Shaoqing, *et al.* Deep residual learning for image recognition [C]// Proc of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 770–778.

[6] Huang Gao, Liu Zhuang, Van Der Maaten L, *et al.* Densely connected convolutional networks [C]// Proc of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 4700–4708.

[7] Xie Saining, Girshick R, Dollár P, *et al.* Aggregated residual transformations for deep neural networks [C]// Proc of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 1492–1500.

[8] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition [EB/OL]. (2015-04-10) [2022-03-28]. https://arxiv. org/pdf/1409. 1556.

[9] Szegedy C, Liu Wei, Jia Yangqing, *et al.* Going deeper with convolutions [C]// Proc of the IEEE conference on Computer Vision and Pattern Recognition. 2015: 1–9.

[10] Tan Mingxing, Le Q. Efficientnet: Rethinking model scaling for convolutional neural networks [C]// Proc of International Conference on Machine Learning. 2019: 6105–6114.

[11] Vaswani A, Shazeer N, Parmar N, *et al.* Attention is all you need [C]// Advances in Neural Information Processing Systems. 2017: 5998–6008.

[12] Wang Xiaolong, Girshick R, Gupta A, *et al.* Non-local neural networks [C]// Proc of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 7794-7803.

[13] Zhao Hengshuang, Jia Jiaya, Koltun V. Exploring self-attention for image recognition [C]// Proc of the IEEE Conference on Computer Vision and Pattern Recognition. 2020: 10076–10085.

[14] Ramachandran P, Parmar N, Vaswani A, *et al.* Studying standalone self-attention in vision models [EB/OL]. (2019-06-13) [2022-03-28]. https://arxiv. org/pdf/1906. 05909.

[15] Carion N, Massa F, Synnaeve G, *et al.* End-to-End object detection with transformers [C]// Proc of European Conference on Computer Vision. 2020: 213-229.

[16] Dosovitskiy A, Beyer L, Kolesnikov A, *et al.* An image is worth 16x16 words: Transformers for image recognition at scale [EB/OL]. (2021-01-03) [2022-03-28]. https://arxiv. org/pdf/2010. 11929.

[17] Chu Xiangxiang, Tian Zhi, Zhang Bo, *et al.* Conditional positional encodings for vision transformers [EB/OL]. (2021-05-18) [2022-03-28]. https://arxiv. org/pdf/2102. 10882.

[18] Han Kai, Xiao An, Wu Enhua, *et al.* Transformer in Transformer [EB/OL]. (2021-10-26) [2022-03-28]. https://arxiv. org/pdf/2103. 00112.

[19] Touvron H, Cord M, Douze M, *et al.* Training data-efficient image transformers & distillation through attention [C]// Proc of International Conference on Machine Learning. 2021, 139: 10347-10357.

[20] Yuan Li, Chen Yunpeng, Wang Tao, *et al.* Tokens-to-token vit: Training vision transformers from scratch on imagenet [C]// Proc of the IEEE/CVF International Conference on Computer Vision. 2021: 558-567.

[21] Wang Wenhai, Xie Enze, Li Xiang, *et al.* Pyramid vision Transformer: A versatile backbone for dense prediction without convolutions [C]// Proc of the IEEE/CVF International Conference on Computer Vision. 2021: 568-578.

[22] Liu Ze, Lin Yutong, Cao Yue, *et al.* Swin Transformer: Hierarchical vision Transformer using shifted windows [C]// Proc of the IEEE/CVF International Conference on Computer Vision. 2021: 10012-10022.

[23] Rao Yongming, Zhao Wenliang, Liu Benlin, *et al.* Dynamicvit: Efficient vision transformers with dynamic token sparsification [EB/OL]. (2021-10-26) [2022-03-28]. https://arxiv. org/pdf/2106. 02034.

[24] Lin Hezheng, Cheng Xing, Wu Xiangyu, *et al.* CAT: Cross attention in vision Transformer [EB/OL]. (2021-06-10) [2022-03-28]. https://arxiv. org/pdf/2106. 05786.

[25] Wu Haiping, Xiao Bin, Codella N, *et al.* Cvt: Introducing convolutions to vision transformers [C]// Proc of the IEEE/CVF International Conference on Computer Vision. 2021: 22-31.

[26] Dai Zihang, Liu Hanxiao, Le Q V, *et al.* Coatnet: Marrying convolution and attention for all data sizes [C]// Advances in Neural Information Processing Systems. 2021, 34: 3965-3977

[27] Chen Zhiyang, Zhu Yousong, Zhao Chaoyang, *et al.* Dpt: Deformable patch-based Transformer for visual recognition [C]// Proc of the 29th ACM International Conference on Multimedia. 2021: 2899-2907.

[28] Zhang Dong, Zhang Hanwang, Tang Jinhui, *et al.* Feature pyramid Transformer [C]// Proc of the European Conference on Computer Vision. 2020: 323-339.

[29] Liu Songtao, Huang Di, Wang Yunhong. Receptive field block net for accurate and fast object detection [C]// Proc of the European Conference on Computer Vision. 2018: 385-400.

[30] Bao Hangbo, Li Dong, Wei Furu, *et al.* Unilmv2: Pseudo-masked language models for unified language model pre-training [C]// Proc of International Conference on Machine Learning. 2020: 642–652.

[31] Hu Han, Gu Jiayuan, Zhang Zheng, *et al.* Relation networks for object detection [C]// Proc of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 3588–3597.

[32] Raffel C, Shazeer N, Roberts A, *et al.* Exploring the limits of transfer learning with a unified text-to-text Transformer [J]. Journal of Machine Learning Research, 2020, 21 (140): 1–67.

[33] Ba J L, Kiros J R, Hinton G E. Layer normalization [EB/OL]. (2016-07-21) [2022-03-28]. https://arxiv. org/pdf/1607. 06450.

[34] Deng Jia, Dong Wei, Socher R, *et al.* Imagenet: A large-scale hierarchical image database [C]// Proc of the IEEE Conference on Computer Vision and Pattern Recognition. 2009: 248–255.

[35] Graham B, El-Nouby A, Touvron H, *et al.* LeViT: A vision Transformer in ConvNet's clothing for faster inference [C]// Proc of the IEEE/CVF International Conference on Computer Vision. 2021: 12259-12269.